J1040 U.S. PTO
10/076116
02/14/02

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of: **SWAMI**

Serial No. **NOT YET ASSIGNED**

Filing Date: **HEREWITH**

For: **AN IMPROVED LOOK-UP TABLE APPARATUS TO PERFORM TWO-BIT ARITHMETIC OPERATION INCLUDING CARRY GENERATION**

I HEREBY CERTIFY THIS PAPER OR FEE IS BEING DEPOSITED WITH THE U.S. POSTAL SERVICE "EXPRESS MAIL POST OFFICE TO ADDRESSEE" SERVICE UNDER 37 CFR 1.10 ON THE DATE INDICATED BELOW AND IS ADDRESSED TO: BOX PATENT APPLICATIONS, ASSISTANT COMMISSIONER FOR PATENTS, WASHINGTON, D.C. 20231.

EXPRESS MAIL NO: EL 768137995 US
DATE OF DEPOSIT: February 14, 2002
NAME: Dawn Kimler
SIGNATURE: Dawn Kimler

#6

TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Director, U.S. Patent and Trademark Office
Washington, D.C. 20231

Sir:

Transmitted herewith is a certified copy of the priority Indian Application No. 151/Del/2001.

Respectfully submitted,

John F. Woodson II

JOHN F. WOODSON, II
Reg. No. 45,236
Allen, Dyer, Doppelt, Milbrath & Gilchrist, P.A.
255 S. Orange Avenue, Suite 1401
Post Office Box 3791
Orlando, Florida 32802
Telephone: 407/841-2330
Fax: 407/841-2343
Attorney for Applicant

GOVERNMENT OF INDIA
MINISTRY OF COMMERCE & INDUSTRY,
PATENT OFFICE, DELHI BRANCH,
W-5, WEST PATEL NAGAR,
NEW DELHI-110 008.

*I the undersigned being an officer duly authorized in accordance with the provision of the Patent Act, 1970 hereby certify that annexed hereto is the true copy of the Application, Complete Specification and Drawing Sheets filed in connection with Application for Patent No.**0151/Del/01** dated **15th February 2001**.*

Witness my hand this 21th day of January, 2002.

(H.C. BAKSHI)
Deputy Controller of Patents & Designs.

FORM 1

THE PATENTS ACT, 1970

(39 of 1970)

APPLICATION FOR GRANT OF A PATENT

(See Sections 5(2), 7, 54 and 135)

1. I/we,

STMicroelectronics Ltd., an Indian Company, of Plot No. 2 & 3, Sector 16 A, Institutional Area, Noida – 201 3001, Uttar Pradesh, India.

2. hereby declare –

(a) that I am/we are in possession of an invention, titled *"An Improved Look Up Table Apparatus To Perform Two Bit Arithmetic Operation Including Carry Generation."*

(b) that the ~~provisional~~/ complete specification relating to this invention is filed with this application

(c) that there is no lawful ground of objection to the grant of a patent to me/us.

3. further declare that the inventor(s) for the said inventions is/are

(i) SWAMI Parvesh, an Indian national, of G-244, Nanak Pura, New Delhi – 110 021, India.

4. I/we claim the priority from the application(s) filed in convection countries, particulars of which are as follows: NA

5. I/we state that the said invention is an improvement in or modification of the invention the particulars of which are as follows and of which I/we are the applicant/patentee: NIL

6. I/we state that the application is divided out of my/our application, the particulars of which are given below and pray that this application be deemed to have been filed on under section 16 of the Act. NIL

7. That I am/we are the assignee or legal representative of the true and first inventors

8. That my/our address for service in India is as follows:

ANAND & ANAND, Advocates
B-41, Nizamuddin East
New Delhi – 110 013

Tel Nos.: (11) 4645078, 4645076, 4640360
Fax Nos.: (11) 4624243, 4642060

ORIGINAL

9- I/We the true and first inventors of this invention or ~~the applicant(s) in the convention country~~ declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a) PARVESH SWAMI , **an Indian National of G-244, Nanak Pura, New Delhi – 110 021, India.**

Signature

Dated this 15th day of February ~~2000~~ 2001

10- that to the best of my/our knowledge, information and belief the fact and matters stated herein are correct and that there is no lawful ground of objection to grant of patent to me/us on this application.

11- Following are the attachment with the application

(a) Complete specification (3 copies)
(b) Abstract
(c) Formal drawings
(d) Power of Attorney
(e) Form 1 (in triplicate)
(f) Form 3 (in duplicate)
(g) Fee Rs. 5000/- In cash/cheque/bank draft bearing no. , date
On Bank.

I/We request that a patent may be granted to me/us for the said invention.

Dated this 15th day of February ~~2000~~ 2001

Signature
STMicroelectronics Limited

To

The Controller of Patents
The Patent Office, Delhi

00-IND-300

Form 2

THE PATENTS ACT, 1970

COMPLETE SPECIFICATION

SECTION 10

15 FEB 2001

0 1 5 1 DEL 0 1

"An Improved Look Up Table Apparatus To Perform Two Bit Arithmetic Operation Including Carry Generation"

ORIGINAL

STMicroelectronics Ltd., Plot No. 2 & 3, Sector 16A, Institutional Area, Noida – 201 3001, Uttar Pradesh, India, an Indian Company

The following specification particularly describes and ascertains the nature of this invention and the manner in which it is to be performed:

The present invention relates to an improved look up table apparatus to perform two bit arithmetic operation including carry generation.

Background of the Invention

Programmable logic devices are known in which programmable look up tables are used to perform any logic function of upto the number of inputs it has. The outputs of such look up tables can be combined together using similar look up tables to provide more complex functions of wide fanin.

Look up tables which are used for performing elementary logic functions can be used for performing some of the special functions like additions, subtraction, counting etc. But the size of one look up table is quite large for peforming these special functions. For doing one bit full addition two look up tables are to be used, one for computing the sum and one for generating the carry. For example four input look up tables which are of very good size for general use but are larger than necessary for one bit full adder or counters. So it is the wastage of resources of look up table to use it for addition or counting. However the counters and adders are very often used in digital logic. If addition or counting of larger number of bits is required then larger number of look up tables are required (two for each bit of operation) and this results in reduced speed of operation and increased wastage of look up table resources.

The US Patent No 5481486 and US Patent 5274581 describe look up tables for use in programmable logic devices, which are modified to facilitate use of these tables to provide adders (including substractors) and various types of counters. The invention in said US patents divide 4 input look up table (LUT) into two 3 input LUT for implementing the arthimetic functions. These implement only one bit full addition because they are using first 3 input LUT for sum generation

bits and other 3 inputs LUT to generate the bits to control the carry logic. So finally they have a 4 input LUT with single output and a carry logic, whereas the instant invention provide 2-bit output with carry out logic using a single 4 input LUT.

Object and summary of the invention:

The object of this invention is to provide the improved ways to implement two bit arithmetic or counters using look up table in the programmable logic devices.

The second object of this invention to use the look up table resources efficiently for implementing the funtions of two inputs or more than two inputs.

Another object of this invention to provide programmable logic devices made up of look up tables in which two bit full addition or counting can be performed without much waste of resources in look up tables and this saves three look up table of four inputs .

Yet another object of this invention to provide ways of improving speed performance of these adders and counters in programmable logic devices made up of look up tables.

To achieve said objectives this invention provides a programmable look up table (LUT) apparatus, which includes a plurality of programmable data storage cells, each of which produces a cell output signal indicative of the data stored in that cell, and means for normally selecting from all of said cell output signals any one of said cell output signals as a normal output signal on a normal output lead of said look-up table apparatus, said means for normally selecting being responsive to a plurality of first input signals such that each of said first input

signals normally controls a respective one of a plurality of successive selection means which collectively comprise said means for selecting, a first said selection means selecting one of two mutually exclusive and collectively exhaustive subsets of said cell output signals, and each succeeding selection means selecting one of two mutually exclusive and collectively exhaustive subsets of the cell output signals selected by the preceding selection means until a final one of said selection means produces said normal output signal on said normal output lead, an **improvement** for enabling said look up table (LUT) apparatus to perform two bit arithmetic operation comprising:

- dividing said LUT apparatus into two equal halves, except final selection means, each half comprising half the remaining selection means, half the number of said data storage cells and half the said input signals,
- a first means for choosing selection input for the final selection means in each said half to be either a first input signals from the second half during normal mode, or the carry output from the previous bit operation during arithmetic mode, while the final selection means at the output of the complete LUT apparatus is a second input signal from the second half, and
- a second means for connecting the output from the final stage of the first half as the least significant bit output of the two bit arithmetic operation and using the output of the final stage of the second half as the most significant bit out of the two bit arithmetic operation during arithmetic mode, while allowing normal selection operation using one of the input signals from the other half during normal mode.

The above programmable look up table (LUT) apparatus further includes means to selectively apply either one input signal of the second half or one input signal of the first half as the first input signal to the second half.

The above programmable look up table (LUT) apparatus further includes additional logic means connected to each half for generating carry out for the corresponding bit operation, while simultaneously generating the sum output, using the same memory elements of said LUT.

The said logic means comprising exclusive-OR of the outputs from the penultimate selection means of said half as a selection signal for selecting either the carry-in signal or the second input signal to said half to generate said carry out signal.

The above programmable look up table (LUT) apparatus further includes a counting mode of operation wherein storage element at the output of each said half are used to store the result of the previous arithmetic operation for use an input to the said half for counting whenever the counting mode is selected.

The said first means in each half is a multiplexing means.

The wherein the select input to said multiplexing means in each half is from a memory.

The said second means in an AND Gate.

The said first means for each half is either the first input signal from the other half during normal mode or the carry output from the lower significant bit operation during arithmetic mode, while the final selection means at the output

of the complete LUT apparatus is the XOR of the last selection signal from each half, thereby enabling the use of said LUT as either a single LUT of 'n' inputs or 2 independent LUTs of 'n-1' inputs, each in normal mode, while retaining all the functionality of the arithmetic mode of operation.

An electronic and counting unit including a LUT, as herein described.

Brief Description of the Drawings

The invention will now be explained with reference to the accompanying drawings

FIG 1 shows a schematic block diagram of prior art look up table apparatus.

FIG 2 shows a schematic block diagram of the modified Look up Table apparatus according to this invention .

FIG 3 shows a schematic block diagram of a complete arithmetic and counting unit using look up table of **FIG 2.**

FIG 4 shows a schematic block diagram of the further modified look up table apparatus of **FIG 2** using the XOR gate.

FIG 5 shows the schematic block diagram of a complete arithmetic and counting unit using the modified look up table of **FIG 4.**

Detailed description of' the drawings:

FIG 1 shows a conventional four input look up table. Look up table **L1** has 16 memory elements **12-1** through **12-16** each of which stores one bit of information.Each memory element (ME) may be a flip-flop, a random access memory(DRAM or SRAM) , EPROM, EEPROM, a cell of a first-in first-out (FIFO), a ferroelectric memory cell, a fuse, an antifuse or the like.The contents

of these memory cells can be fixed or they can be programmed once or repeatedly. The four inputs A-D select the output of one of the memory elements to pass to the output of look up table **L1**. The output of each memory element is applied to one of inputs of respective AND gates

14-1 through **14-16** . Input A is applied to other input of AND gates **14-2, 14-4, 14-6, 14-8, 14-10, 14-12** and**14-14,** and , after inversion is applied to other inputs of respective AND gates **14-1, 14-3, 14-5, 14-7, 14-9, 14-11, 14-13,** and **14-15**. Accordingly half of AND gates **14** are enabled by input A and other half is disabled. OR gates **16** pass the outputs of enabled AND gates **14** to next level of AND gates **18**.

Input B is applied to one of the inputs of AND gates **18-2, 18-4, 18-6** and **18-8,** and, inversion is applied to one of the inputs of AND gates **18-1, 18-3, 18-5** and **18-7.** Input B enables half of the AND gates **18** and disables the second half of those AND gates. Input B therefore selects four of the eight memory cells **12** outputs selected by input A. OR gates **20** pass the outputs of enabled AND gates to next level of AND gates **22**.

Input C is applied to one of the inputs of AND gates **22-2** and **22-4**, and, inversion is applied to one of the inputs of AND gates **22-1** and **22-3**. Input C enables one half of AND gates **22** and disables the second half of those AND gates. Thus input C selects two of four memory cells **12** outputs selected by input B. OR gates **24** pass the outputs of enabled AND gates to next level of AND gates **26** .

Input D is applied to one of the inputs of AND gate **26-2,** and , inversion is applied to one of inputs of AND gate **26-1**. Input D enables one half of the AND gates **26** and disables second half of those AND gates. Thus input D selects one

of the two memory cells **12** outputs selected by input C. OR gate **28** passes the output of enabled AND gate as final output of look up table **L1**.

Two look up tables are required for a full adder(Here this term refers to both adder and subtractor). One for sum generation and one for carry generation. With some modifications in the prior art look up table one look up table can be used for implementing two full adders. Accordingly,look up table **L1** is modified in accordance with this invention as shown in **FIG 2** and Fig4 so that it can provide two sum outputs on leads OUT0 and OUT1 with OUT0 as LSB and OUT1 as MSB and outputs required to generate carry on leads C_L and C_Li corresponding to LSB and C_U and C_Ui corresponding to MSB.

LUT apparatus of Fig1 is modified in **FIG** 2for 2 bit arthimetic operation including carry generation by splitting D input into D and Di and C input is split into C and Ci. Output of sixteen memory cells **12** are connected to one of the inputs of respective AND gates **14- 1A** through **14-16A**. Input A is applied to one of inputs of AND gates **14-2A, 14-4A, 14-6A,** and **14-8A,** and its inversion is applied to one of the inputs of AND gates **14-1A, 14-3A, 14-5A,**and **14-7A.** Input Ci is applied to one of inputs of AND gates **14-10A, 14- 12A, 14-14A,** and **14-16A** and its inversion is applied to one of the inputs of AND gates **14-9A, 14-11A, 14-13A,** and **14-15A**. Thus inputs A and Ci enable half of the AND gates **14A** and disable the other half of those AND gates. OR gates **16A** pass the output of enabled AND gates **14A** to the next level of AND gates **18A**.

Input B is applied to one of inputs of AND gates **18-2A** and **18-4A,** and, its inversion is applied to one of the inputs of AND gates **18-1A** and **18-3A**. Input Di is applied to one of inputs of AND gates **18-6A,** and, its inversion is applied to one of the inputs of AND gates **18-5A** and **18-7A**. Inputs B and Di enable half of the AND gates **18A** and disable the other half of those AND gates. Thus

inputs B and Di select four of eight memory cells **12** outputs selected by inputs A and Ci. OR gates **20A** pass the outputs of enabled AND gates **18A** to next level of AND gates **22A** and to the outputs C_L , C_Li, C_U, C_Ui of look up table **L2A**.

The output of switch **26-1A** which selects one of the inputs Cin or C and is controlled by output of ME **172A** is applied to one of inputs of AND gate **22-2A** and its inversion is applied to one of the inputs of AND gate **22-1A.** The output of switch **26-2A** which selects one of the inputs CY0 or C and is controlled by output of ME **172A** is applied to one of inputs of AND gate **22-4A** and its inversion is applied to one of the inputs of AND gate **22-3A**. Two of three inputs (either Cin and CY0 or C) enable half of the AND gates **22A** and disable the other half of those AND gates. Thus two of four memory cells **12A** outputs selected by inputs B and Di are selected by abovementioned two of four inputs. OR gates 24A pass the outputs of enabled AND gates **22A** to next level of AND gates **28A**. Output of OR gate **24A** is passed to the next level AND gates **28A** and output of OR gate **24-2A** is also passed to the output OUT1 of look up table **L2A**.

The output of AND gate **30A** whose inputs are input D and output of ME **172A** is applied to one of inputs of AND gate **28-2A** and its inversion is applied to one of the inputs of AND gate **28-1A**. The abovementioned output enables half of the AND gates **28A** and disables the other half of those AND gates. Thus one of the two memory cells **12** outputs passed by previous level OR gates **24A** is selected. OR gate **32A** pass the outputs of enabled AND gates **28A** to final output OUT0 of look up table **L2A**.

FIG 3 shows how the modified look up table **L2A** of **FIG 2** can be used with other circuitry in accordance with this invention to provide highly flexible and

powerful logic block for use in programmable logic arrays. Programmable logic block (PLB) as shown in **FIG 3** has four regular data inputs A_arith - D_arith (These inputs are confgurably connected to A-D inputs respectively of look up table), carry in input CYin which is the carry out output of another PLB, add_sub input which can dynamically set the addition or subtraction mode during binary arithmetic operation or up or down counting mode during binary counter operation.

PLB of **FIG 3** has five outputs, four regular data outputs from output drivers and carry out output. The carry out output connects to the carry in input of another PLB, typically adjacent PLB and is used for carrying out addition, subtraction, addition and subtraction , or counting (up, down, up and down, skip). Skip counting here means that while counting states can be skipped just by giving the value by which it should be skipped(both up and down).

When PLB is used to perform normal logic operation rather than addition, subtraction or counting, switch **11-1A**, which is controlled by output of ME **170A**, connects A_arith input of PLB to A input of look up table **L2A,** switches **11-2A** and **11-3A,** which are controlled by output of ME **171A** and **172A** respectively connect A_arith input of PLB to Ci input of look up table, switch **26-1A** of look up table **L2A** passes C_arith input of PLB to its output , switch **26-2A** of look up table **L2A** also passes C_arith input of PLB to its output and AND gate **30A** of look up table **L2A** passes D_arith input of PLB to its output. Switch **11-4A** , which is controlled by output of ME **172A**, connects B_arith input of PLB to Di input of look up table.The outputs OUT0 and OUT1 of look up table **L2A** are applied to outputs of PLB OUT0 and OUT1 respectively.These outputs OUT0 and OUT1 of look up table **L2A** are also connected to the inputs of flip-flops **19-1A** and **19-2A** respectively to get registered outputs Q0 and Q1 respectively.

In normal mode of operation of PLB two functions of two inputs (these two inputs are A_arith and B_arith ,and, C_arith and D_arith) can also be implemented using same look up table **L2A**. In this mode all the connections remain the same as in normal mode (explained in last para) except that switch **26- 1A** of look up table **L2A** passes Cin input of look up table **L2A** to its output and the output of AND gate **30A** of look up table **L2A** is tied to logic low.

In arithmetic mode of operation one PLB can perform maximum of two places of binary addition or subtraction or addition and subtraction . In this mode all the connections are same as that in the mode explained in last para. The outputs C_L, C_Li (which in this mode is inversion of C_L) and C_U, C_Ui (inversion of C_U) of look up table **L2A** are connected to inputs of switches **17-1A** and **17-2A** respectively which are controlled by output of OR gate **21A** . OR gate **21A** has add_sub input of PLB and output of ME **175A** connected to its inputs. The switches **17A** implement XOR functionality in this mode where second input to them is complement of first input. Output of switch **17-1A** is connected to control input of switch **15-1A**, whose inputs are outputs of gates **13-1A** and **13-2A**, which is used to generate carry out signal CY0 . The output of gate **13-1A** is passed to output of switch **15- 1A** when its control input is logic low. Gate **13-2A** has input CYin, output of ME **174A** as its inputs. Gate **13-1A** has input B_arith, output of ME **173A** as its inputs. In this mode gate **13-2A** can be configured to pass either carry from previous stage or logic low signal and gate **13-1A** is configured to pass B_arith signal . Output Cin of gate **13-2A** is connected to Cin input of look up table **L2A**. Output of switch **17-2A** is connected to one of the inputs of gate **22A** and other input of this gate is connected to output of ME **172A.** The output of gate **22A** controls switch **15-2A** which generates the carryout signal CYout. In adder and counter modes AND gate **22A** passes output of switch **17-2A** and in normal mode it passes

logic low value which maps D_arith input from the general routing matrix onto the carry chain. The switch **15-2A** has input D_arith and CY0 output of switch **15-1A** as its inputs and D_arith is selected when its control input is at logic low value. Thus output of switch **15-2A** generates signal CYout which is carry output of PLB. For two bit arithmetic operation D_arith and B_arith are taken as augend (for addition or minuend for subtraction and where D_arith is MSB) and C_arith and A_arith are taken as addend or subtrahend (where C_arith is MSB). The sum outputs (where output acts as MSB) are passed directly as outputs of PLB and they can be registered as explained in the normal mode of operation. While performing addition, output of OR gate **21A** is tied to logic low value, in subtraction mode this output is tied to logic high and in addition and subtraction mode add_sub input signal is passed through the OR gate **21A** whose other input is ME **175A** which controls the additions and subtraction functions . Whenever one full addition is required MEs **12-9A** through **12- 16A** can be configured so as to pass CY0 to the output Cyout / OUT1 of look up table **L2A**.

In counter mode of operation of PLB the configuration is same as that explained in last para with some minor changes. The switch **11-1** passes the output Q0 of flip-flop **19-1A** to its output thus connecting Q0 to input A of look up table **L2A**. Similarly switches **11-2A** and **11-3A** pass the output Q1 of flip-flop **19-2A** to its output so as to connect Q1 to input Ci of look up table **L2A**. Gate **13-1A** can pass either input B or can pull its output high. So if it is first stage of counter then it is pulled high otherwise it passes B to its output. Gate **13-2A** passes the carry of previous stage to its output or pulls down its output to logic low value. If it is first stage of counter then it passes a logic low value to its output otherwise it passes the previous carry. The input add_sub can be used as up/down control just the same way it is used for addition/subtraction. For doing subtraction OR gate **21A** is configured to pulls its output to logic high value.

In skip counting mode of operation the configuration is same as that in counter mode with some minor changes. In this mode we provide difference of the value of next state and the current state as inputs to PLB and this differece is provided at inputs B_arith and D_arith. In this case gate **13-1A** always passes input B_arith to its output.

This architecture is very useful for implementing normal functions(4 I/P functions) and arithmatic functions with less resources.

Since more than one switches are controlled by a single ME (e.g ME 172 controls 6 switches), so by providing independent MEs to different switches , we can get more flexible architecture.

FIG 4 is a further modification of **FIG2** by providing XOR gate for upper and lower part selection by splitting A input into A and Ai and C input is split into C and Ci. Output of sixteen memory cells **12** are connected to one of the inputs of respective AND gates **14- 1B** through **14-16B**. Input Ai is applied to one of inputs of AND gates **14-2B, 14-4B, 14-6B,** and **14-8B,** and its inversion is applied to one of the inputs of AND gates **14-1B, 14-3B, 14-5B,**and **14-7B**. Input Ci is applied to one of inputs of AND gates **14-10B, 14- 12B, 14-14B,** and **14-16B** and its inversion is applied to one of the inputs of AND gates **14-9B, 14-11B, 14-13B,** and **14-15B**. Thus inputs Ai and Ci enable half of the AND gates **14B** and disable the other half of those AND gates. OR gates **16B** pass the output of enabled AND gates **14B** to the next level of AND gates **18B**.

Input B is applied to one of inputs of AND gates **18-2B** and **18-4B,** and its inversion is applied to one of the inputs of AND gates **18-1B** and **18-3B**. Input D is applied to one of inputs of AND gates **18-6B** and **18-8B,** and, its inversion is applied to one of the inputs of AND gates **18-5B** and **18-7B**. Inputs B and C

enable half of the AND gates **18B** and disable the other half of those AND gates. Thus inputs B and D select four of eight memory cells **12** outputs selected by inputs Ai and Ci. OR gates **20** pass the outputs of enabled AND gates **18B** to next level of AND gates **22B** and to the outputs C_L , C_Li, C_U, C_Ui of look up table **L2B**.

The output of switch **26-1B** which selects one of the inputs Cin or C and is controlled by output of ME **172B** is applied to one of inputs of AND gate **22-2B** and its inversion is applied to one of the inputs of AND gate **22-1B.** The output of switch **26-2** which selects one of the inputs CY0 or B and is controlled by output of ME **172B** is applied to one of inputs of AND gate **22-4B** and its inversion is applied to one of the inputs of AND gate **22-3B**. Two of four inputs (either Cin and CY0 or C and B) enable half of the AND gates **22B** and disable the other half of those AND gates. Thus two of four memory cells **12** outputs selected by inputs B and C are selected by abovementioned two of four inputs. OR gates **24B** pass the outputs of enabled AND gates **22B** to next level of AND gates **28B**. Output of OR gate **24B** is passed to the next level AND gates **28B** and output of OR gate **24-2B** is also passed to the output OUT1 of look up table **L2B**.

The output of AND gate **32B** whose inputs are XOR **30B** of inputs A and D , and output of ME **172B** is applied to one of inputs of AND gate **28-2B** and its inversion is applied to one of the inputs of AND gate **28-1B**. The abovementioned output enables half of the AND gates **28B** and disables the other half of those AND gates. Thus one of the two memory cells **12** outputs passed by previous level OR gates **24B** is selected. OR gate **34B** pass the outputs of enabled AND gates **28B** to final output OUT0 of look up table **L2B**.

FIG 5 shows how the modified look up table **L2B** of **FIG 4** can be used with other circuitary in accordance with this invension to provide highly flexible and powerful logic block for use in programmable logic arrays. Programmable logic block (PLB) as shown in **FIG 5** has four regular data inputs A_arith - D_arith (These inputs are configurably connected to A-D inputs respectively of look up table), carry in input CYin which is the carry out output of another PLB, add_sub input which can dynamically set the addition or subtraction mode during binary arithmetic operation or up or down counting mode during binary counter operation.

PLB of **FIG 5** has five outputs, four regular data outputs from output drivers and carry out output. The carry out output connects to the carry in input of another PLB, typically adjacent PLB and is used for carrying out addition, subtraction, addition and subtraction, or counting (up, down, up and down, skip). Skip counting here means that while counting states can be skipped just by giving the value by which it should be skipped(both up and down).

When PLB is used to perform normal logic operation rather than addition, subtraction or counting, switch **11-1B**, which is controlled by output of ME **170B**, connects A_arith input of PLB to Ai input of look up table **L2B,** switch **11-2B,** which is controlled by output of ME **171B**, connects C_arith input of PLB to Ci input of look up table, switch **26-1B** of look up table **L2B** passes C_arith input of PLB to its output , switch **26-2B** of look up table **L2B** passes B_arith input of PLB to its output and AND gate **32B** of look up table **L2B** passes the XOR of A_arith and D_arith inputs of PLB. The outputs OUT0 and OUT1 of look up table **L2B** are applied to outputs of PLB OUT0 and OUT1 respectively.These outputs OUT0 and OUT1 of look up table **L2B** are also connected to the inputs of flip-flops **19-1B** and **19-2B** respectively to get registered outputs Q0 and Q1 respectively.

In normal mode of operation of PLB two functions of two inputs (these two inputs are A_arith and B_arith ,and, C_arith and D_arith) can also be implemented using same look up table **L2B**. In this mode all the connections remain the same as in normal mode (explained in last para) except that switch **26- 1B** of look up table **L2B** passes Cin input of look up table **L2** to its output, switch **26-2B** of look up table **L2B** passes CY0 input of look up table **L2B** to its output and the output of AND gate **32B** of look up table **L2B** is tied to logic low.

In arithmetic mode of operation one PLB can perform maximum of two places of binary addition or subtraction or addition and subtraction . In this mode all the connections are same as that in the mode explained in last para. The outputs C_L , C_Li (which in this mode is inversion of C_L) and C_U, C_Ui (inversion of C_U) of look up table **L2B** are connected to inputs of switches **17-1B** and **17-2B** respectively which are controlled by output of OR gate **21B** . OR gate **21B** has add_sub input of PLB and output of ME **175B** connected to its inputs. The switches **17B** implement XOR functionality in this mode where second input to them is complement of first input. Output of switch **17-1B** is connected to control input of switch **15-1B**, whose inputs are outputs of gates **13-1B** and **13-2B**, which is used to generate carry out signal CY0 . The output of gate **13-1B** is passed to output of switch **15- 1B** when its control input is logic low. Gate **13-1B** has input B_arith, output of ME **173B** as its inputs. Gate **13-2B** has input CYin , output of ME **174B** as its inputs. In this mode gate **13-2B** can be configured to pass either carry from previous stage or logic low signal and gate **13-1B** is configured to pass B_arith signal . Output Cin of gate **13-2B** is connected to Cin input of look up table **L2B**. Output of switch **17-2B** is connected to one of the inputs of gate **22B** and other input of this gate is connected to output of ME **172B.** The output of gate **22B** controls switch **15- 2B**

which generates the carryout signal CYout. In adder and counter modes AND gate **22B** passes output of switch **17-2B** and in normal mode it passes logic low value which maps D_arith input from the general routing matrix onto the carry chain. The switch **15-2B** has input D_arith and CY0 output of switch **15-1B** as its inputs and D_arith is selected when its control input is at logic low value. Thus output of switch **15-2B** generates signal CYout which is carry output of PLB. For two bit arithmetic operation D_arith and B_arith are taken as augend (for addition or minuend for subtraction and where D_arith is MSB) and C_arith and A_arith are taken as addend or subtrahend (where C_arith is MSB). The sum outputs (where output acts as MSB) are passed directly as outputs of PLB and they can be registered as explained in the normal mode of operation. While performing addition, output of OR gate **21B** is tied to logic low value, in subtraction mode this output is tied to logic high and in addition and subtraction mode add_sub input signal is passed through the OR gate **21B** whose other input is ME **175B** which controls the additions and subtraction functions . Whenever one full addition is required MEs **12-9B** through **12- 16B** can be configured so as to pass Cyout / CY0 to the output OUT1 of look up table **L2B**.

The arithmetic operation comprises addition, subtraction and counting.

In counter mode of operation of PLB the configuration is same as that explained in last para with some minor changes. The switch **11-1B** passes the output Q0 of flip-flop **19-1B** to its output thus connecting Q0 to input Ai of look up table **L2B**. Similarly switch **11-2B** passes the output Q1 of flip-flop **19-2B** to its output so as to connect Q1 to input Ci of look up table **L2B**. Gate **13-1B** can pass either input B or can pull its output high. So if it is first stage of counter then it is pulled high otherwise it passes B to its output. Gate **13-2B** passes the carry of previous stage to its output or pulls down its output to logic low value. If it is first stage of counter then it passes a logic low value to its output

otherwise it passes the previous carry. The input add_sub can be used as up/down control just the same way it is used for addition/subtraction. For doing subtraction OR gate **21B** is configured to pulls its output to logic high value.

In skip counting mode of operation the configuration is same as that in counter mode with some minor changes. In this mode we provide difference of the value of next state and the current state as inputs to PLB and this differece is provided at inputs B_arith and D_arith. In this case gate **13-1B** always passes input B_arith to its output.

This architecture is very useful for implementing normal functions(4 I/P functions) and arithmatic functions with less resources& good speed.

Since more than one switches are controlled by a single ME (e.g ME 172 controls 4 switches), so by providing independent MEs to different switches, we can get more flexible architecture. e.g if all these 4 switches are controlled by independent ME , then this architecture will be able to implement two functions of three inputs with two common inputs (B & C)

CLAIMS:

1. In programmable look up table (LUT) apparatus, which includes a plurality of programmable data storage cells, each of which produces a cell output signal indicative of the data stored in that cell, and means for normally selecting from all of said cell output signals any one of said cell output signals as a normal output signal on a normal output lead of said look-up table apparatus, said means for normally selecting being responsive to a plurality of first input signals such that each of said first input signals normally controls a respective one of a plurality of successive selection means which collectively comprise said means for selecting, a first said selection means selecting one of two mutually exclusive and collectively exhaustive subsets of said cell output signals, and each succeeding selection means selecting one of two mutually exclusive and collectively exhaustive subsets of the cell output signals selected by the preceding selection means until a final one of said selection means produces said normal output signal on said normal output lead, an **improvement** for enabling said look up table (LUT) apparatus to perform two bit arithmetic operation comprising:
   - dividing said LUT apparatus into two equal halves, except final selection means, each half comprising half the remaining selection means, half the number of said data storage cells and half the said input signals,
   - a first means for choosing selection input for the final selection means in each said half to be either a first input signals from the second half during normal mode, or the carry output from the previous bit operation during arithmetic mode, while the final selection means at the output of the complete LUT apparatus is a second input signal from the second half, and

- a second means for connecting the output from the final stage of the first half as the least significant bit output of the two bit arithmetic operation and using the output of the final stage of the second half as the most significant bit out of the two bit arithmetic operation during arithmetic mode, while allowing normal selection operation using one of the input signals from the other half during normal mode.

2. The programmable look up table (LUT) apparatus as claimed in claim 1 further including means to selectively apply either one input signal of the second half or one input signal of the first half as the first input signal to the second half.

3. The programmable look up table (LUT) apparatus as claimed in claim 1 further including additional logic means connected to each half for generating carry out for the corresponding bit operation, while simultaneously generating the sum output, using the same memory elements of said LUT.

4. The programmable look up table (LUT) apparatus as claimed in claim 3 wherein said logic means comprising exclusive-OR of the outputs from the penultimate selection means of said half as a selection signal for selecting either the carry-in signal or the second input signal to said half to generate said carry out signal.

5. The programmable look up table (LUT) apparatus as claimed in claim 1 further including a counting mode of operation wherein storage element at the output of each said half are used to store the result of the previous

arithmetic operation for use an input to the said half for counting whenever the counting mode is selected.

6. The programmable look up table (LUT) apparatus as claimed in claim 1 wherein said first means in each half is a multiplexing means.

7. The programmable look up table (LUT) apparatus as claimed in claim 6 wherein the select input to said multiplexing means in each half is from a memory.

8. The programmable look up table (LUT) apparatus as claimed in claim 1 wherein said second means in an AND Gate.

9. The programmable look up table (LUT) apparatus as claimed in claim 1 wherein the said first means for each half is either the first input signal from the other half during normal mode or the carry output from the lower significant bit operation during arithmetic mode, while the final selection means at the output of the complete LUT apparatus is the XOR of the last selection signal from each half, thereby enabling the use of said LUT as either a single LUT of 'n' inputs or 2 independent LUTs of 'n-1' inputs, each in normal mode, while retaining all the functionality of the arithmetic mode of operation.

10. Electronic and counting unit including a LUT as herein described.

11. The programmable look up table (LUT) apparatus substantially as herein described with reference to and as illustrated in the accompanying drawings.

Dated this 15th day of February, 2001

Shanti Kumar

of **Anand & Anand, Advocates**
Agents for the Applicants

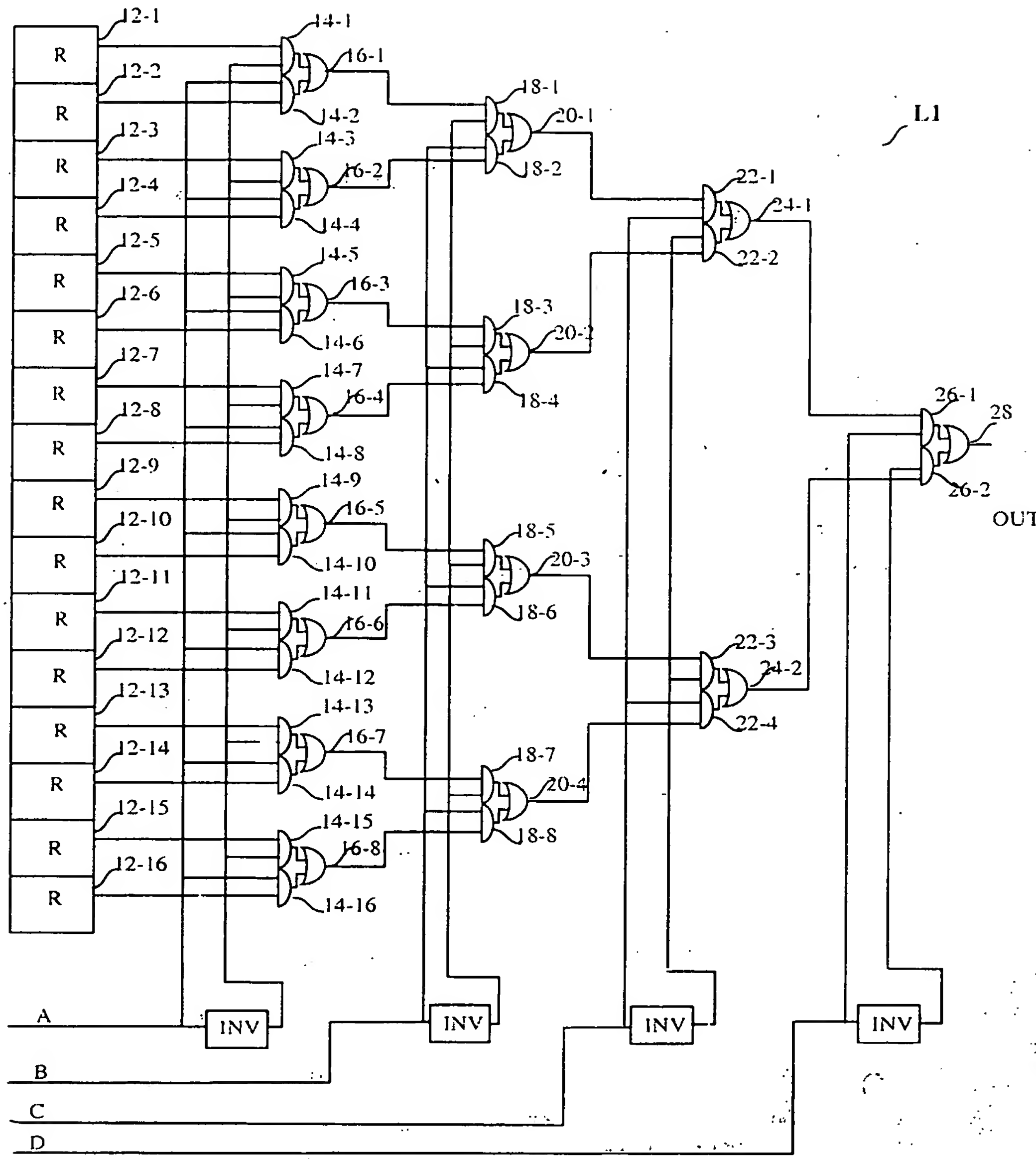

FIG1
Prior Art

Shanti Kumar

of Anand And Anand Advocates
Attorney for the Applicants

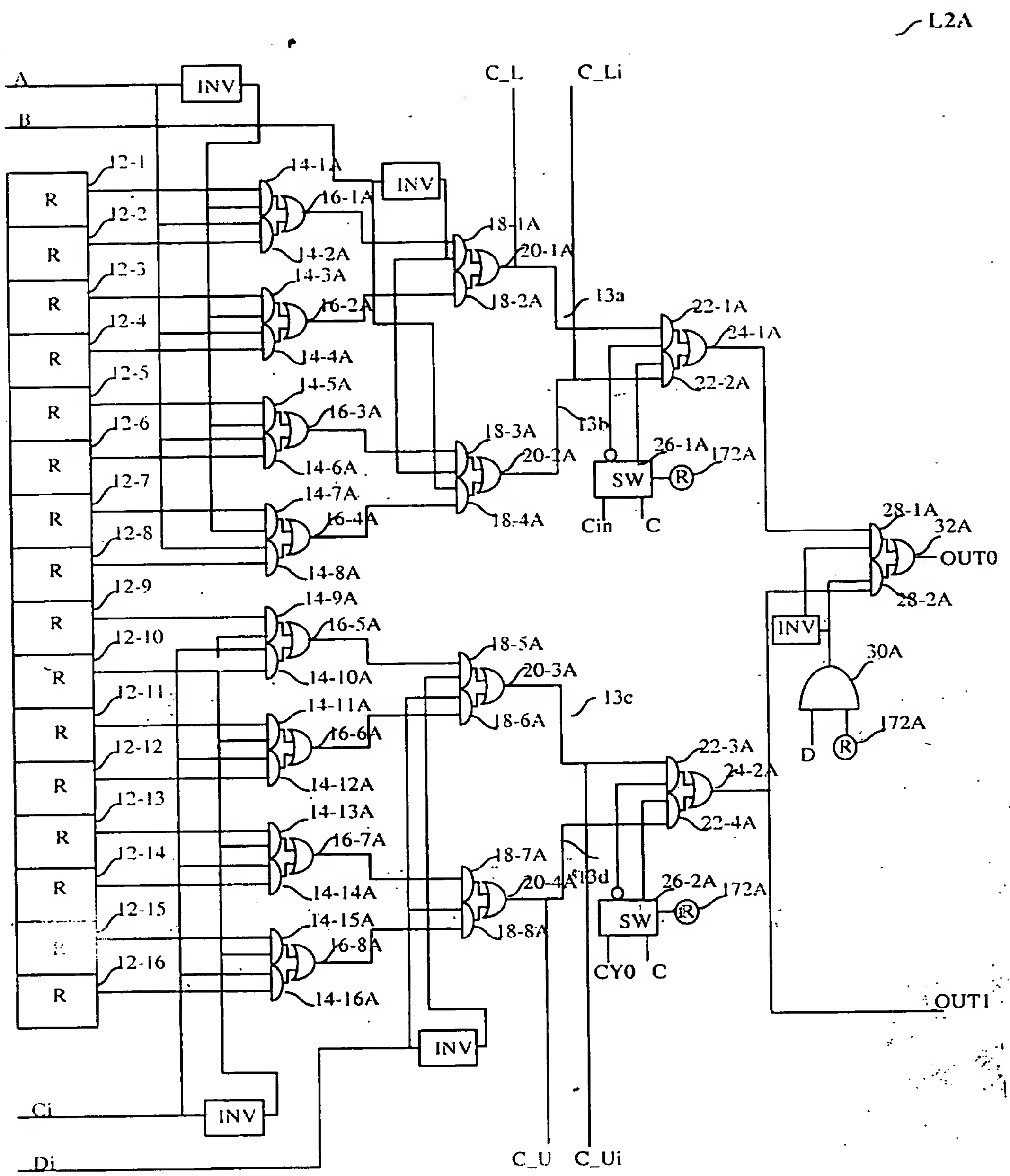

FIG2

Shanti Kumar

of Anand And Anand Advocates
Attorney for the Applicants

STMicroelectronics Ltd
No.

5 Sheets
Sheet

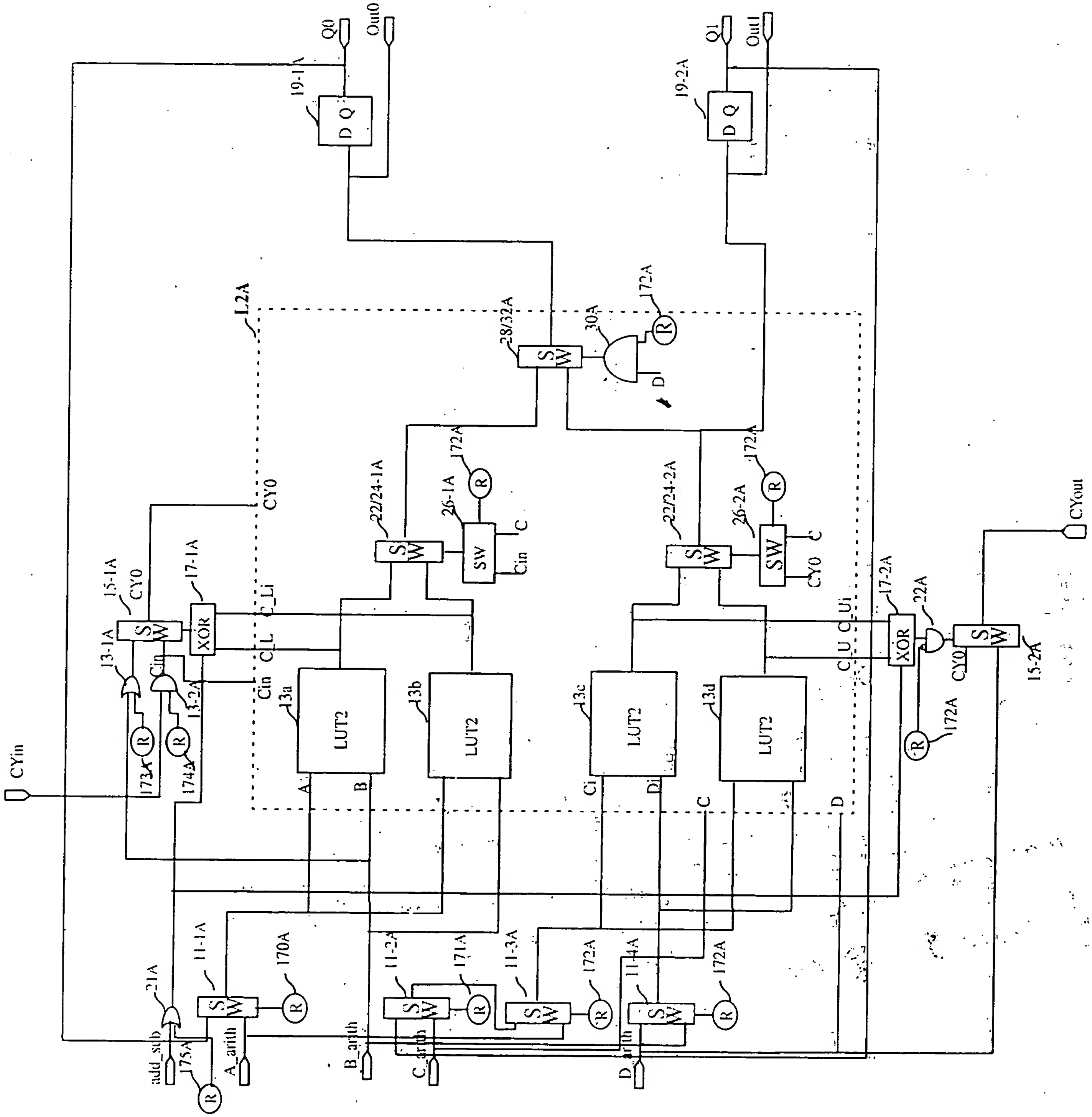

FIG 3

Shanti Kumar

of Anand And Anand Advocates
Attorney for the Applicants

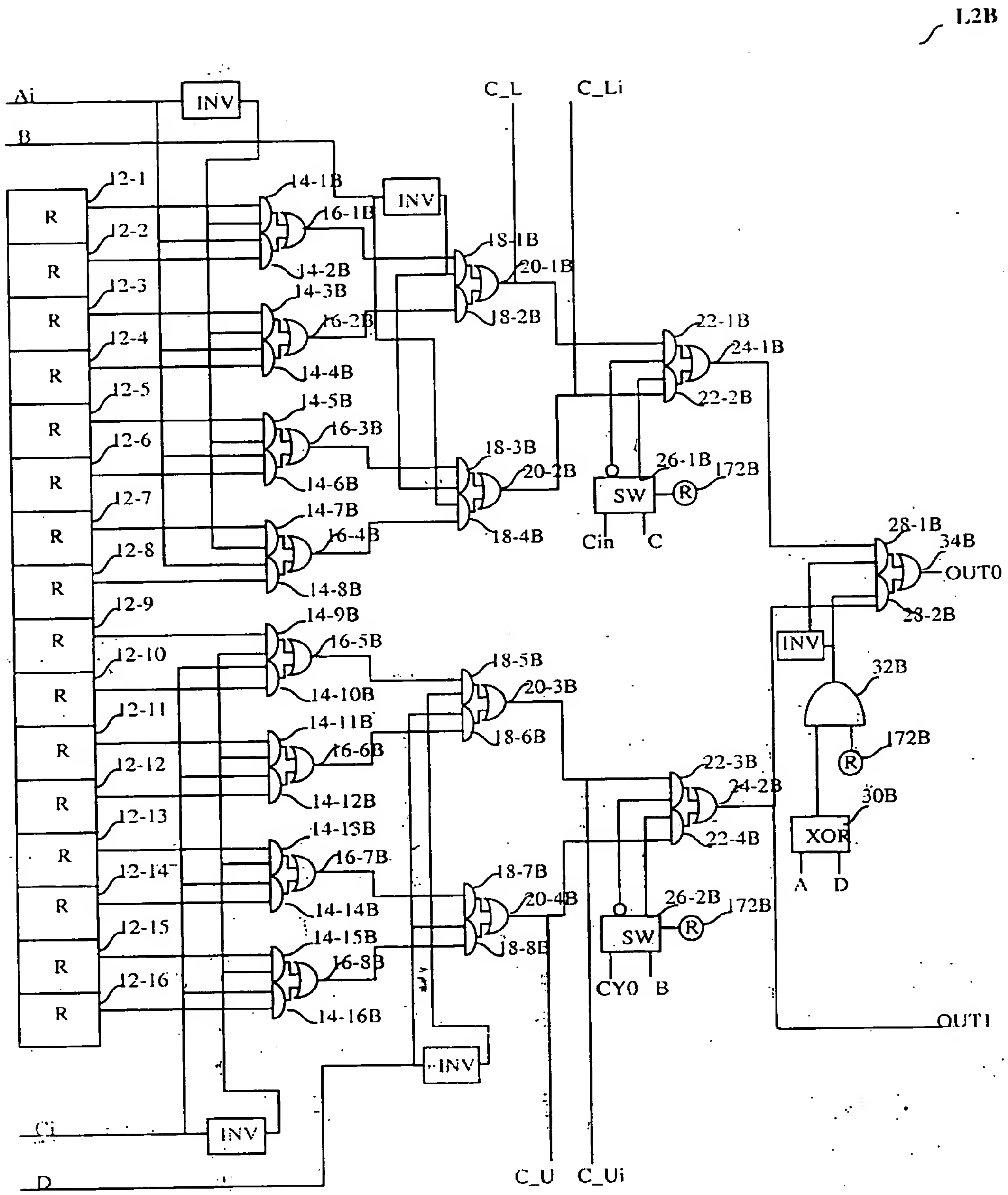

FIG4

Shanti Kumar

of Anand And Anand Advocates
Attorney for the Applicants

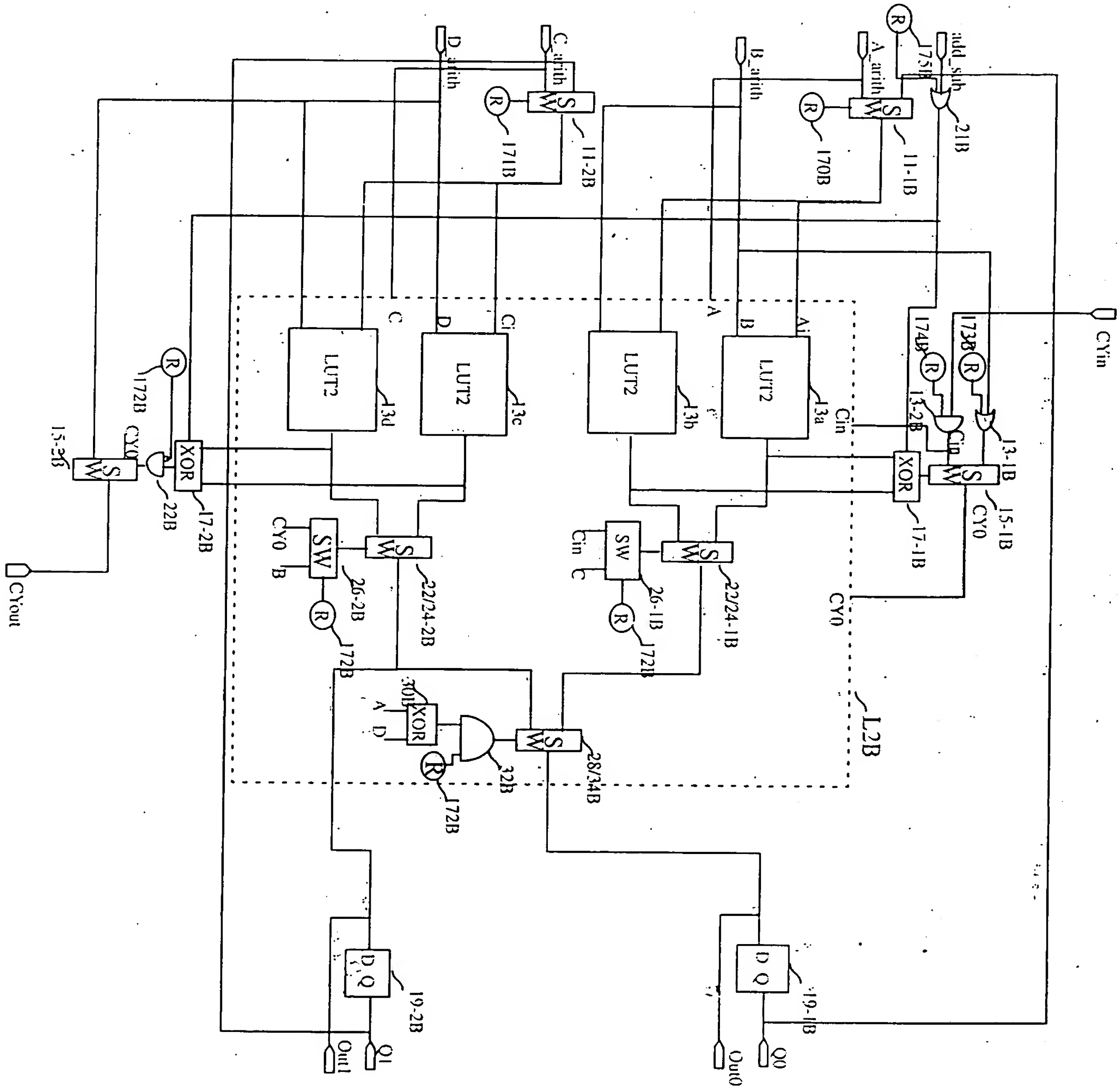

FIG5

Shanti Kumar

of Anand And Anand Advocates
Attorney for the Applicants